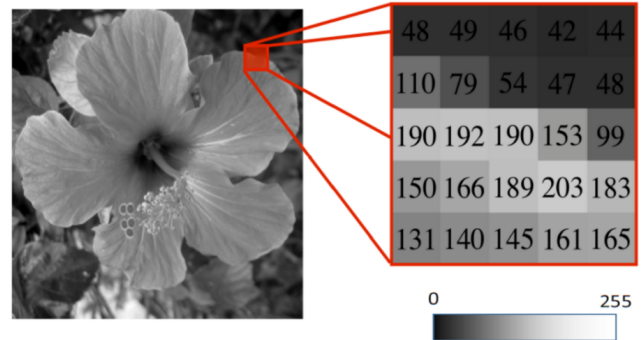


Se rendre à l'adresse suivante : <https://mybinder.org/v2/gh/CDERYCKE/SNT-Photo/master> et démarrer le TP 3.

1. Compréhension

Pour détecter si le pixel situé en (x, y) appartient ou non à un contour, une des méthodes consiste à chercher une rupture d'intensité entre 2 pixels symétriques (diamétralement opposés) par rapport à $P(x, y)$.

On travaille avec une image en niveaux de gris et on s'intéresse au pixel de valeur 79 dont les coordonnées sont (x, y) .



- 1) Compléter la position de chaque pixel composant l'image suivante :

	$P(x, y)$	
		$P(x+1, y+1)$

- 2) On calcule alors la différence (soustraction) de niveaux de gris entre ces pixels symétriques

	Pixel "79" :	Pixel "166" :
	Calcul à faire :	
	En Python : difference = Niveau_Gris (..... ,) - Niveau_Gris (..... ,)	
	Avec le carré :	

- 3) Pour ne pas avoir une différence négative, il suffit d'élever au carré le résultat.
Reprendre les calculs précédents en élevant la différence au carré.
- 4) Parmi ces deux pixels, lequel semble appartenir à un éventuel contour. Pourquoi ?
- 5) A partir de quelle valeur (de quel seuil) pourrait-on décider qu'un pixel appartient ou non à un contour ?
(Faire une proposition qui sera par la suite testée dans le programme Python).

On va effectuer le calcul suivant :

6) Quel seuil de la norme choisir pour séparer le pixel “54” qui semble faire partie d’un contour du pixel “166” ?
(Faire une proposition qui sera ensuite testée dans le programme Python).

2. Programmation en Python de la détection de contours d'une image

```
for y in range(1,hauteur-1):
    for x in range(1,.....):
# a, b, c et d contiennent les valeurs du niveau de gris des pixels voisins du
pixel P(x,y)
        a = img_gris.getpixel((x-1, y-1))
        b = img_gris.getpixel((x+1, y+1))
        c = img_gris.getpixel((x-1, y+1))
        d = img_gris.getpixel((x+1, y-1))

# partie à modifier
        norme = .....
        seuil = ..... # choix du seuil pour la détection de
contours

# le pixel (x,y) fait parti du contour si sa norme est supérieure au seuil et on
le colorie en noir. Sinon, on le colorie en blanc.
        if norme > seuil : # test conditionnel si .... alors .....
            img_contour.putpixel((x, y), (.....,.....,.....))
        else :
            img_contour.putpixel((x, y), (.....,.....,.....))
```

2